

Remarks:

Claims 1, 3-4, and 34-38 were currently pending with claims 1, 34 and 37 being independent.

No claims have been amended, added or cancelled by way of this paper and, therefore, claims 1, 3-4, and 34-38 are currently pending with claims 1, 34 and 37 being independent.

The Rejection of Claims under 35 USC § 102

Claims 1, 3, 4 and 34-38 were rejected under 35 U.S.C. 102(e) as being anticipated U.S. Patent 7,025,975 to Pazandak et al. ("Pazandak"). Applicant believes the Examiner mistakenly referenced Patent number 7,025,975 as the "Ryan" patent. The references made by the Examiner are consistent with statements in Pazandak and Applicant therefore assumes the discrepancy is a typographical error and the foregoing remarks are based upon this assumption.

Applicant respectfully disagrees that Pazandak anticipates the claims because Pazandak: 1) does not receive queries in SQL; 2) does not interpret the queries . . . by associating at least one declarative language function; and 3) does not convert queries represented by declarative language queries into a plurality of Java statements. Because each of these limitations may be found in claims 1 and 34, Pazandak cannot anticipate or render the present claims obvious. Claims 3 - 4 and 35 - 38 depend respectively from these independent claims, are believed to add additional patentable subject matter, and are allowable for at least the reasons stated above.

Pazandak Does Not Receive Queries in SQL

Applicant respectfully disagrees that Pazandak discloses "receiving queries on a pervasive computing device in SQL." Pazandak teaches "light-weight guided [Natural Language Interface] client interfaces." Col.3, ll.18-19. A Natural Language Interface (NLI) is "is a particular mode or method of user interaction with a computing device or devices in which the user employs natural language, such as the English language and grammar, to communicate inputs, for example, queries and commands, to the devices and applications of the devices." Col. 1, ll. 27-33. Thus, Pazandak *receives* its queries in a natural language, such as English.

This interpretation is consistent with the Figures showing query input, for example Figs. 7-11 and 16, and with various textual examples of input queries, for example:

1. "Find dresses size 18 color . . . " col. 32, ll. 63
2. "Find airports in McLean Tex. where a 747 can land and which have sufficient fuel . . . " col. 33, ll. 4-6
3. "Record program-on channel 11-on Saturday from 3 pm to 4 pm" (or simply "Record-Xena") col. 34, ll.59-61
4. "Water-the front yard-starting at 9 pm-for an hour-and-the back yard-starting at 10 pm-for 30 minutes-each night-but not if-Internet weather-indicates it will rain that day-or if-local water rationing is-in effect" col. 34, ll. 62-66.

Furthermore, this interpretation is consistent with Figures 14 and 17 showing a interface for mapping between natural language elements and database objects for a menu, and a translation into SQL from a natural language query, respectively.

Additionally, if Pazandak received queries in SQL, it would be effectively useless. One of the many disclosed uses of the Pazandak invention is to query a database. Therefore, if Pazandak allowed entry of SQL and simply queried a database using SQL it performs no functions other than passing the SQL query to a database and returning the results. Obviously, Pazandak must do more than this.

***Pezandak does not Interpret the Queries by Associating at
Least One Declarative Language Function.***

Pezandak does not “*interpret* the queries . . . by associating at least one declarative language function” (emphasis added). Instead, Pezandak’s parse tree “can be *externalized, represented* in Lisp notation, the W3C Document Object Model, or XML or by some other means” (emphasis added). In other words, the generated parse tree may be “externalized”, and by externalized Pezandak likely means saved as a text file, in one of many forms. Simply because a parse tree can be externalized in one of several forms does not imply that the externalized forms correspond to the way it internally “*interpret[s]* the queries.”

Furthermore, Pezandak discloses that *grammar rules* and *lexicons* may be encoded in Lisp syntax, but grammar rules and lexicons are not the same as “interpret[ing] queries.” The grammar defines valid combinations of words or elements to form a sentence. The lexicon includes the words of the language. The grammar and lexicon may be defined in any way without respect to the method used to interpret the queries. Commonly grammars are defined rules such as “ $V \rightarrow w$ ”, where V is a nonterminal symbol and w contains terminals and/or nonterminals. However, a computer program using grammar using rules of the form “ $V \rightarrow w$ ” would not likely *interpret* a statement using such a

construction. Instead, the rules would be changed into a more efficient intermediate form. Thus, simply because Pezandak discloses that grammar rules and lexicons may be encoded in Lisp, Pezandak does not disclose that queries are *interpreted* in Lisp. Therefore, Pezandak does not disclose *interpreting* the queries by associating at least one declarative language function.

Similarly, because the queries are not represented by at least one declarative language function, Pezandak cannot disclose “converting the queries *represented by at least one declarative language function* to a plurality of JAVA statements” because the presence of a declarative language function is a condition precedent to the claim limitation.

Conclusion

For at least the reasons set forth above, applicant respectfully submits that claims 1, 3-4, and 34-38 are now in allowable condition and requests a Notice of Allowance. In the event of further questions, the Examiner is urged to call the undersigned. Any additional fee which is due in connection with this amendment should be applied against our Deposit Account No. 19-0522.

Respectfully submitted,

By: /Scott R. Brown/
Scott R. Brown, Reg. No. 40,535
HOVEY WILLIAMS LLP
10801 Mastin Blvd., Suite 1000
84 Corporate Woods
Overland Park, KS66210
(913) 647-9050

ATTORNEYS FOR APPLICANT(S)